

AluraStoreLatam

October 24, 2025

1 Informe de Análisis - Challenge AluraStore Latam

1.0.1 Autor

Jair Garduño Rodriguez

Este proyecto forma parte del desafío **Practicando Python para Data Science** dentro de mi formación en Data Science en ONE. Se enfoca en el análisis de los datos de la tienda Alura, donde, mediante Jupyter Notebook, se han explorado patrones y tendencias, generando visualizaciones que proporcionan insights valiosos para la toma de decisiones.

1.0.2 Importación de datos

```
[178]: # Librerías
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import folium as fl

# Carga de datos
url = "https://raw.githubusercontent.com/alura-es-cursos/
↳challenge1-data-science-latam/refs/heads/main/base-de-datos-challenge1-latam/
↳tienda_1%20.csv"
url2 = "https://raw.githubusercontent.com/alura-es-cursos/
↳challenge1-data-science-latam/refs/heads/main/base-de-datos-challenge1-latam/
↳tienda_2.csv"
url3 = "https://raw.githubusercontent.com/alura-es-cursos/
↳challenge1-data-science-latam/refs/heads/main/base-de-datos-challenge1-latam/
↳tienda_3.csv"
url4 = "https://raw.githubusercontent.com/alura-es-cursos/
↳challenge1-data-science-latam/refs/heads/main/base-de-datos-challenge1-latam/
↳tienda_4.csv"

tienda = pd.read_csv(url)
```

```

tienda2 = pd.read_csv(url2)
tienda3 = pd.read_csv(url3)
tienda4 = pd.read_csv(url4)

tienda.head()

```

```

[178]:

```

	Producto	Categoría del Producto	Precio	Costo de envío	\
0	Asistente virtual	Electrónicos	164300.0	6900.0	
1	Mesa de comedor	Muebles	192300.0	8400.0	
2	Juego de mesa	Juguetes	209600.0	15900.0	
3	Microondas	Electrodomésticos	757500.0	41000.0	
4	Silla de oficina	Muebles	335200.0	20200.0	

	Fecha de Compra	Vendedor	Lugar de Compra	Calificación	\
0	16/01/2021	Pedro Gomez	Bogotá	4	
1	18/05/2022	Beatriz Morales	Medellín	1	
2	15/03/2021	Juan Fernandez	Cartagena	1	
3	03/05/2022	Juan Fernandez	Cali	4	
4	07/11/2020	Maria Alfonso	Medellín	5	

	Método de pago	Cantidad de cuotas	lat	lon
0	Tarjeta de crédito	8	4.60971	-74.08175
1	Tarjeta de crédito	4	6.25184	-75.56359
2	Tarjeta de crédito	1	10.39972	-75.51444
3	Nequi	1	3.43722	-76.52250
4	Nequi	1	6.25184	-75.56359

2 Análisis de facturación

```

[179]: print(f"Ingreso total tienda 1: {tienda.Precio.sum()}")

```

Ingreso total tienda 1: 1150880400.0

```

[180]: print(f"Ingreso total tienda 2: {tienda2.Precio.sum()}")

```

Ingreso total tienda 2: 1116343500.0

```

[181]: print(f"Ingreso total tienda 3: {tienda3.Precio.sum()}")

```

Ingreso total tienda 3: 1098019600.0

```

[182]: print(f"Ingreso total tienda 4: {tienda4.Precio.sum()}")

```

Ingreso total tienda 4: 1038375700.0

3 Ventas por categoría

```
[183]: # Tienda 1
tienda.groupby('Categoría del Producto')['Cantidad de cuotas'].sum().
↳sort_values(ascending=False)
```

```
[183]: Categoría del Producto
Muebles                1446
Electrónicos           1341
Electrodomésticos      906
Juguetes               872
Deportes y diversión   796
Instrumentos musicales  573
Libros                 520
Artículos para el hogar 489
Name: Cantidad de cuotas, dtype: int64
```

```
[184]: # Tienda 2
tienda2.groupby('Categoría del Producto')['Cantidad de cuotas'].sum().
↳sort_values(ascending=False)
```

```
[184]: Categoría del Producto
Electrónicos           1333
Muebles                1265
Electrodomésticos      890
Juguetes               876
Deportes y diversión   720
Instrumentos musicales  659
Libros                 611
Artículos para el hogar 493
Name: Cantidad de cuotas, dtype: int64
```

```
[185]: # Tienda 3
tienda3.groupby('Categoría del Producto')['Cantidad de cuotas'].sum().
↳sort_values(ascending=False)
```

```
[185]: Categoría del Producto
Muebles                1408
Electrónicos           1262
Juguetes               937
Electrodomésticos      882
Deportes y diversión   873
Instrumentos musicales  580
Artículos para el hogar 542
Libros                 517
Name: Cantidad de cuotas, dtype: int64
```

```
[186]: # Tienda 4
tienda4.groupby('Categoría del Producto')['Cantidad de cuotas'].sum().
↳sort_values(ascending=False)
```

```
[186]: Categoría del Producto
Muebles                1392
Electrónicos           1293
Juguetes                921
Deportes y diversión    786
Electrodomésticos       756
Libros                  591
Artículos para el hogar 590
Instrumentos musicales   500
Name: Cantidad de cuotas, dtype: int64
```

4 Calificación promedio de la tienda

```
[187]: print("Tienda 1")
print(f"Calificaciones promedio por cliente: {round(tienda.Calificación.
↳mean(),2)} de {tienda.Calificación.max()}")
```

Tienda 1
Calificaciones promedio por cliente: 3.98 de 5

```
[188]: print("Tienda 2")
print(f"Calificaciones promedio por cliente: {round(tienda2.Calificación.
↳mean(),2)} de {tienda2.Calificación.max()}")
```

Tienda 2
Calificaciones promedio por cliente: 4.04 de 5

```
[189]: print("Tienda 3")
print(f"Calificaciones promedio por cliente: {round(tienda3.Calificación.
↳mean(),2)} de {tienda3.Calificación.max()}")
```

Tienda 3
Calificaciones promedio por cliente: 4.05 de 5

```
[190]: print("Tienda 4")
print(f"Calificaciones promedio por cliente: {round(tienda4.Calificación.
↳mean(),2)} de {tienda4.Calificación.max()}")
```

Tienda 4
Calificaciones promedio por cliente: 4.0 de 5

5 Productos más y menos vendidos

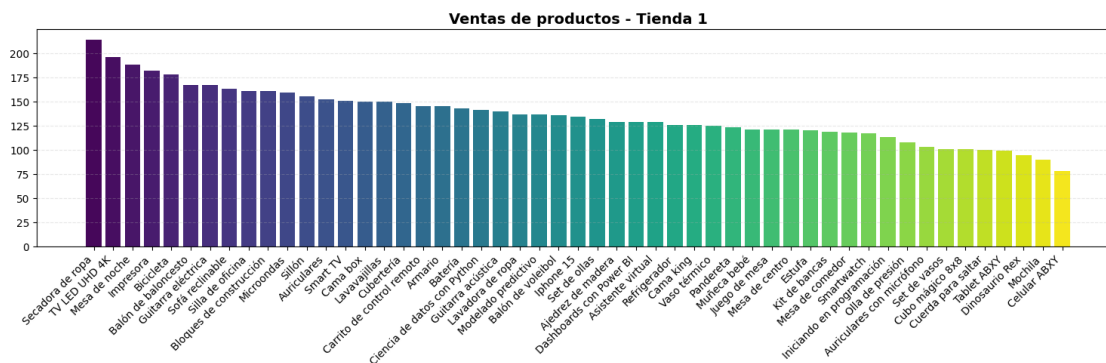
```
[191]: # Tienda 1
# Datos agrupados
ventas = (tienda.groupby('Producto')['Cantidad de cuotas'].sum().
    ↪sort_values(ascending=False).reset_index())

# Paleta moderna (gradiente)
colors = sns.color_palette("viridis", len(ventas))

# Gráfico
plt.figure(figsize=(15,5))
bars = plt.bar(ventas['Producto'], ventas['Cantidad de cuotas'], color=colors)

plt.title('Ventas de productos - Tienda 1', fontsize=14, weight='bold')
plt.xticks(rotation=45, ha='right')
plt.ylabel('')
plt.xlabel('')
plt.grid(axis='y', linestyle='--', alpha=0.3)

plt.tight_layout()
plt.savefig("figures/venta_productos_t1.png", dpi=300, bbox_inches='tight') #_
    ↪PNG de alta calidad
plt.show()
```



En la **Tienda 1**, los productos con mayor demanda corresponden a: - **Secadora de ropa** - **TV LED UHD 4K** - **Mesa de noche**

Por otro lado, los artículos con **menor rotación de ventas** fueron: - **Celular ABXY** - **Mochila** - **Dinosaurio Rex**

```
[192]: # Tienda 2
# Datos agrupados
```

```

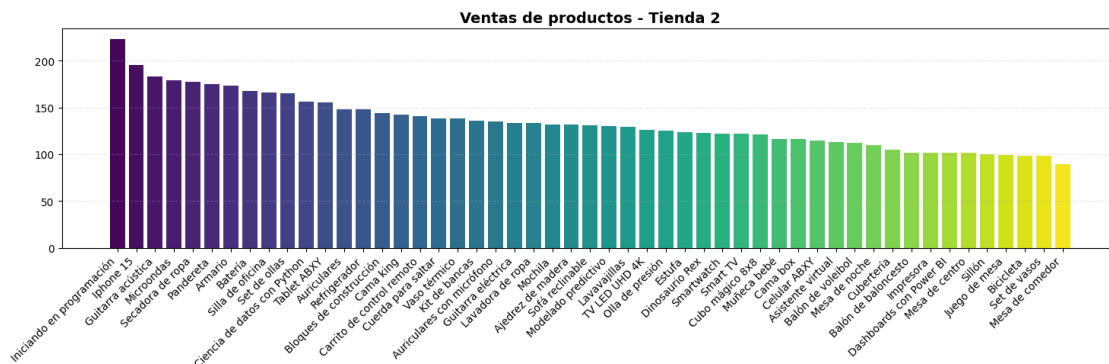
ventas = (tienda2.groupby('Producto')['Cantidad de cuotas'].sum().
    ↪sort_values(ascending=False).reset_index())

# Paleta moderna (gradiente)
colors = sns.color_palette("viridis", len(ventas))

# Gráfico
plt.figure(figsize=(15,5))
bars = plt.bar(ventas['Producto'], ventas['Cantidad de cuotas'], color=colors)

plt.title('Ventas de productos - Tienda 2', fontsize=14, weight='bold')
plt.xticks(rotation=45, ha='right')
plt.ylabel('')
plt.xlabel('')
plt.grid(axis='y', linestyle='--', alpha=0.3)
plt.tight_layout()
plt.savefig("figures/venta_productos_t2.png", dpi=300, bbox_inches='tight') # ↪
    ↪PNG de alta calidad
plt.show()

```



En la **Tienda 2**, los productos con mayor demanda corresponden a: - **Iniciando en programación** - **iPhone 15** - **Guitarra acústica**

Por otro lado, los artículos con **menor rotación de ventas** fueron: - **Mesa de comedor** - **Set de vasos** - **Bicicleta**

```

[193]: # Tienda 3
# Datos agrupados
ventas = (tienda3.groupby('Producto')['Cantidad de cuotas'].sum().
    ↪sort_values(ascending=False).reset_index())

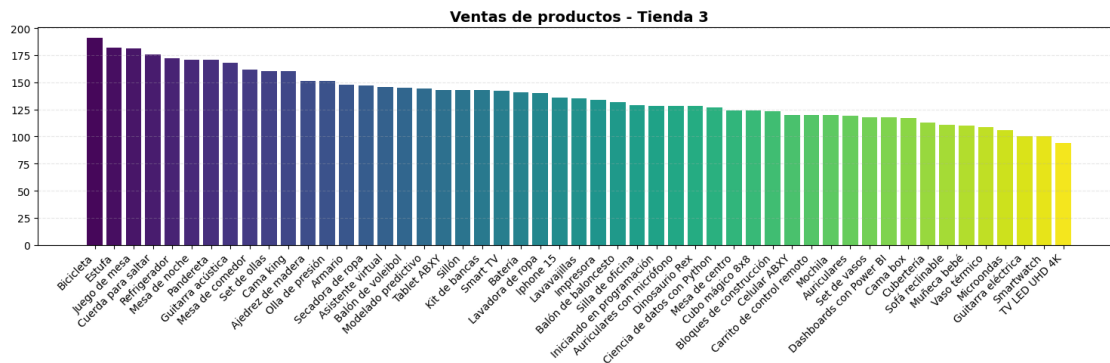
# Paleta moderna (gradiente)
colors = sns.color_palette("viridis", len(ventas))

```

```
# Gráfico
plt.figure(figsize=(15,5))
bars = plt.bar(ventas['Producto'], ventas['Cantidad de cuotas'], color=colors)

plt.title('Ventas de productos - Tienda 3', fontsize=14, weight='bold')
plt.xticks(rotation=45, ha='right')
plt.ylabel('')
plt.xlabel('')
plt.grid(axis='y', linestyle='--', alpha=0.3)
plt.tight_layout()
plt.savefig("figures/venta_productos_t3.png", dpi=300, bbox_inches='tight') #_
    ↪PNG de alta calidad

plt.show()
```



En la **Tienda 3**, los productos con mayor demanda corresponden a: - **Bicicleta** - **Estufa** - **Juego de mesa**

Por otro lado, los artículos con **menor rotación de ventas** fueron: - **TV LED UHD 4K** - **Smartwatch** - **Guitarra eléctrica**

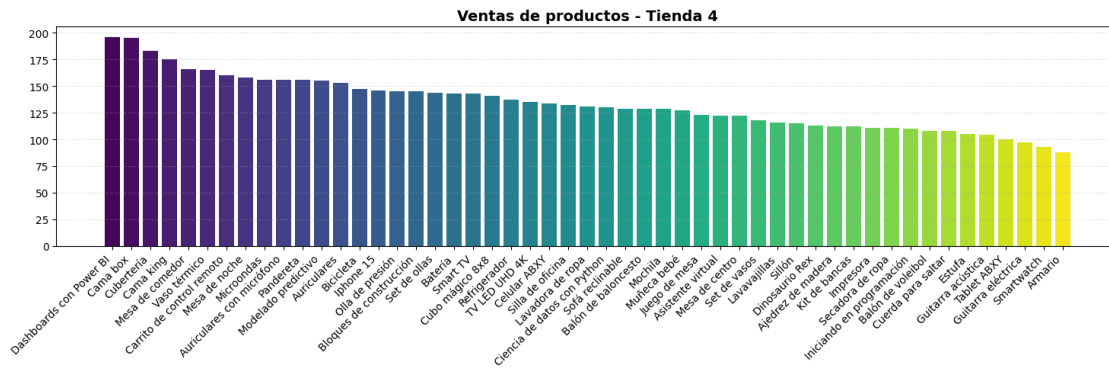
```
[194]: # Tienda 4
# Datos agrupados
ventas = (tienda4.groupby('Producto')['Cantidad de cuotas'].sum()).
    ↪sort_values(ascending=False).reset_index()

# Paleta moderna (gradiente)
colors = sns.color_palette("viridis", len(ventas))

# Gráfico
plt.figure(figsize=(15,5))
bars = plt.bar(ventas['Producto'], ventas['Cantidad de cuotas'], color=colors)

plt.title('Ventas de productos - Tienda 4', fontsize=14, weight='bold')
```

```
plt.xticks(rotation=45, ha='right')
plt.ylabel('')
plt.xlabel('')
plt.grid(axis='y', linestyle='--', alpha=0.3)
plt.tight_layout()
plt.savefig("figures/venta_productos_t4.png", dpi=300, bbox_inches='tight') #
↳PNG de alta calidad
plt.show()
```



En la **Tienda 4**, los productos con mayor demanda corresponden a: - **Dashboards con Power BI** - **Cama box** - **Cubernetaria**

Por otro lado, los artículos con **menor rotación de ventas** fueron: - **Armario** - **Smartwatch** - **Guitarra eléctrica**

6 Envío promedio por tienda

```
[195]: print("Tienda 1")
print(f"El costo promedio de envío es: {round(tienda['Costo de envío'].
↳mean(),2)}")
```

Tienda 1

El costo promedio de envío es: 26018.61

```
[196]: print("Tienda 2")
print(f"El costo promedio de envío es: {round(tienda2['Costo de envío'].
↳mean(),2)}")
```

Tienda 2

El costo promedio de envío es: 25216.24

```
[197]: print("Tienda 3")
print(f"El costo promedio de envío es: {round(tienda3['Costo de envío'].
↳mean(),2)}")
```


Tienda 3

El costo promedio de envío es: 24805.68

```
[198]: print("Tienda 4")
print(f"El costo promedio de envío es: {round(tienda4['Costo de envío'].
↪mean(),2)}")
```

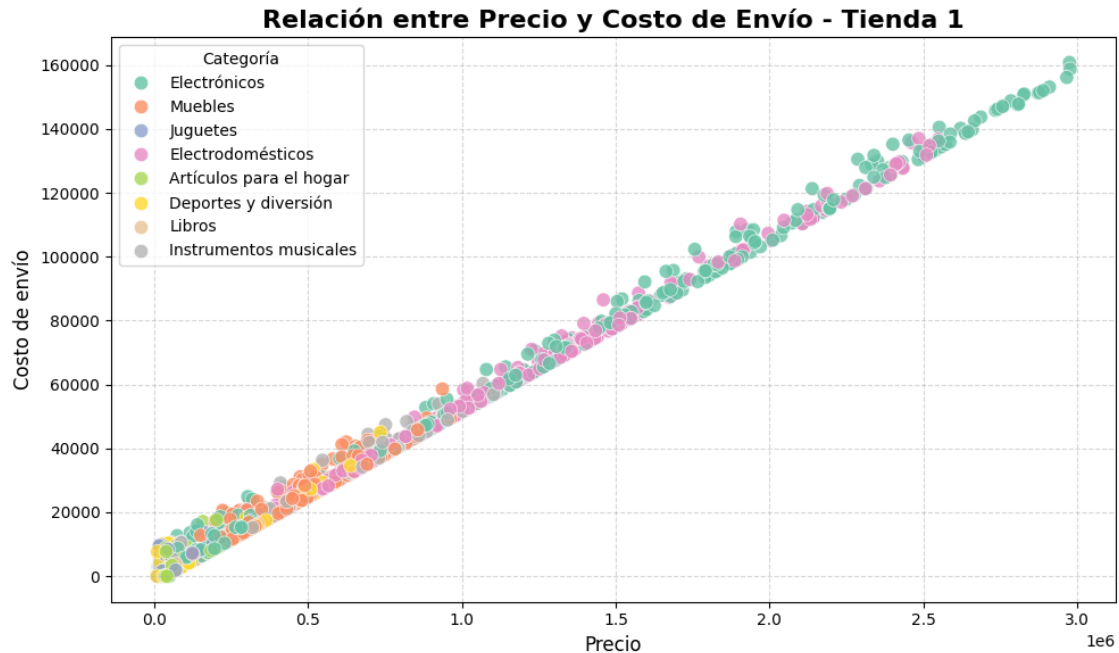
Tienda 4

El costo promedio de envío es: 23459.46

7 Relación entre precio y costo de envío

Como podemos observar en el siguiente gráfico de dispersión, existe una relación positiva entre el precio de los productos y el costo de envío en la Tienda 1. A medida que aumenta el precio del producto, también tiende a aumentar el costo de envío.

```
[199]: plt.figure(figsize=(10,6))
sns.scatterplot(
    data=tienda,
    x='Precio', y='Costo de envío',
    hue='Categoría del Producto',
    palette='Set2',
    s=80, alpha=0.8,
    edgecolor='w', linewidth=0.5
)
plt.title('Relación entre Precio y Costo de Envío - Tienda 1', fontsize=16,
↪weight='bold')
plt.xlabel('Precio', fontsize=12)
plt.ylabel('Costo de envío', fontsize=12)
plt.legend(title='Categoría')
plt.grid(True, linestyle='--', alpha=0.5)
plt.tight_layout()
plt.savefig("figures/corr_precio_costo.png", dpi=300, bbox_inches='tight') #
↪PNG de alta calidad
plt.show()
```



```
[200]: print("Tienda 1")
print(f"La correlación entre Precio y Costo de envío es: {tienda['Precio'].corr(tienda['Costo de envío'])}")
```

Tienda 1

La correlación entre Precio y Costo de envío es: 0.9978023817368026

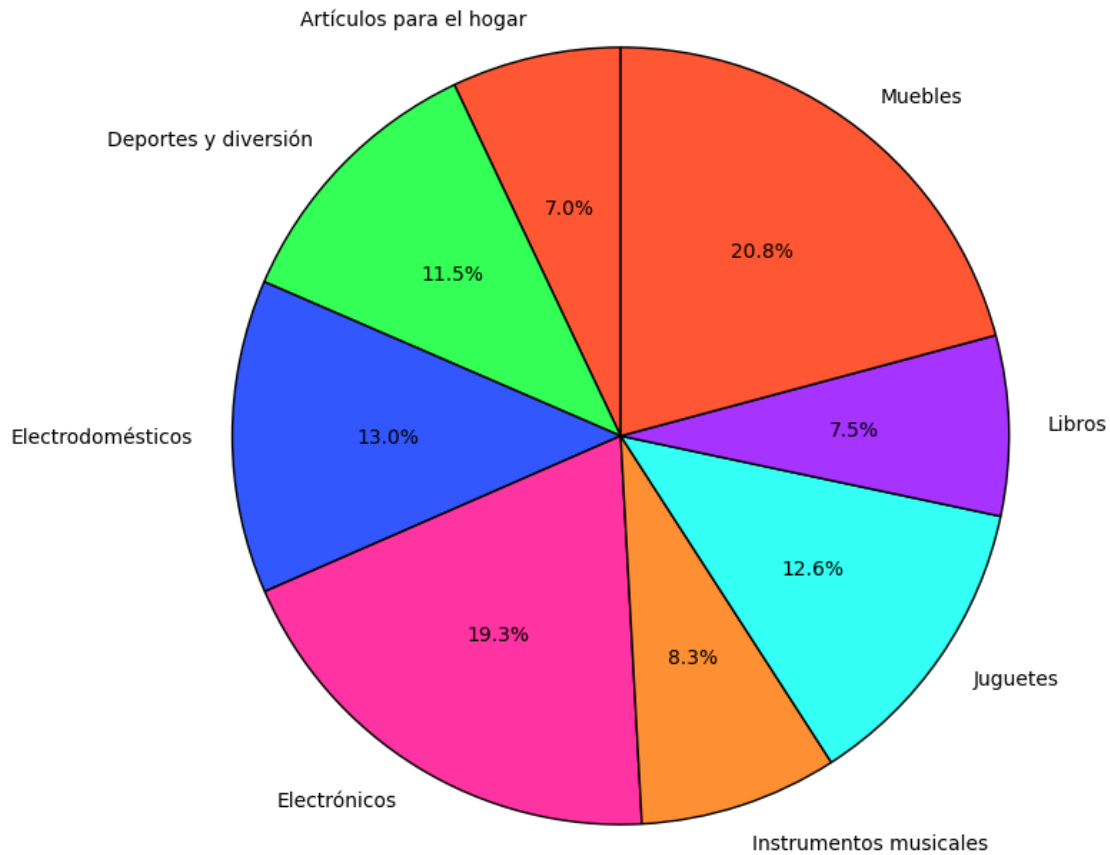
8 Ventas mensuales

```
[201]: # Grafico de ventas mensuales - Tienda 1
colores_brillosos = ['#FF5733', '#33FF57', '#3357FF', '#FF33A1', '#FF8F33',
                    '#33FFFF', '#A633FF']
ventas_por_categoria = tienda.groupby('Categoría del Producto')['Cantidad de_
cuotas'].sum()
ventas_por_categoria.plot.pie(
    autopct='%1.1f%%',
    startangle=90,
    figsize=(8,8),
    colors=colores_brillosos,
    wedgeprops={'edgecolor': 'black'}
)

plt.title('Ventas por Categoría - Tienda 1', fontsize=14, weight='bold')
plt.ylabel('')
plt.tight_layout()
```

```
plt.savefig("figures/ventas_categoria.png", dpi=300, bbox_inches='tight') #  
↳PNG de alta calidad  
plt.show()
```

Ventas por Categoría - Tienda 1



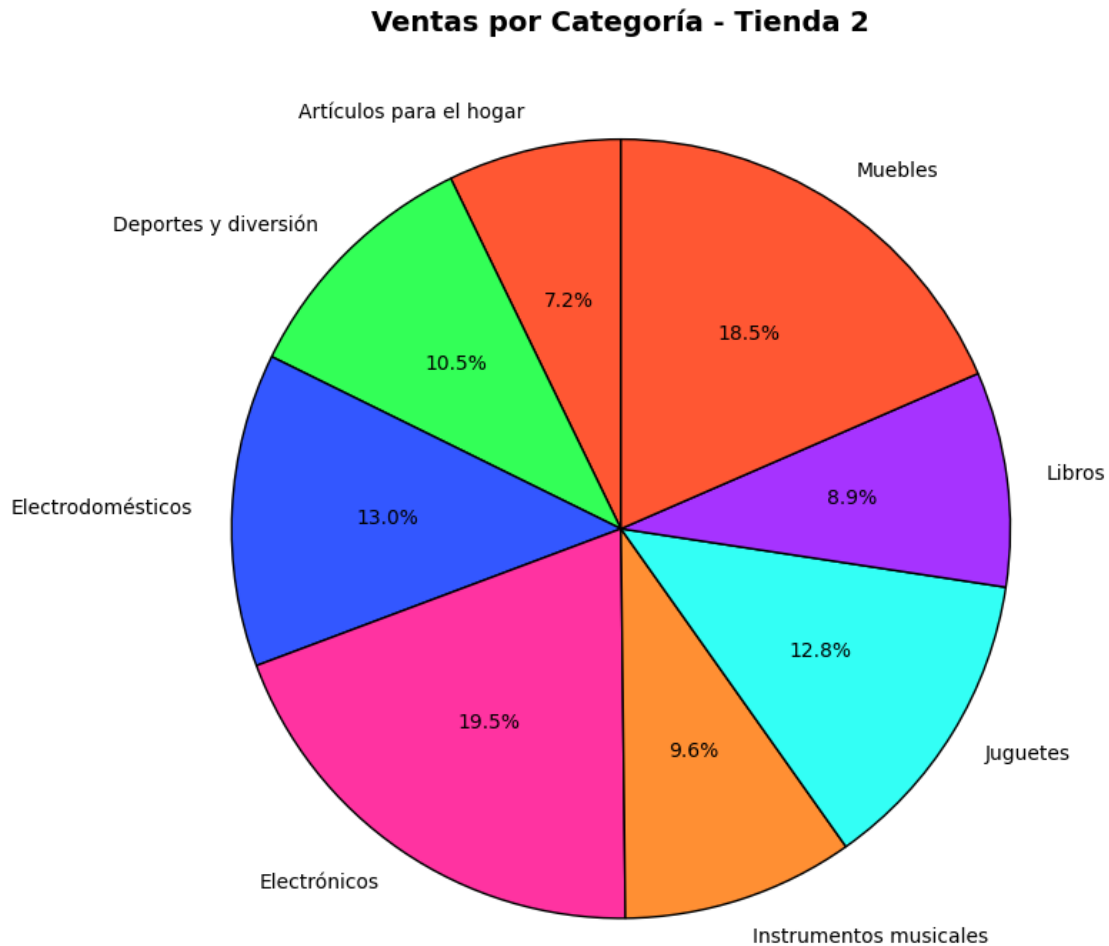
```
[202]: # Grafico de ventas mensuales - Tienda 2  
colores_brillosos = ['#FF5733', '#33FF57', '#3357FF', '#FF33A1', '#FF8F33',  
↳'#33FFF5', '#A633FF']  
ventas_por_categoria = tienda2.groupby('Categoría del Producto')['Cantidad de  
↳cuotas'].sum()  
ventas_por_categoria.plot.pie(  
    autopct='%1.1f%%',  
    startangle=90,  
    figsize=(8,8),  
    colors=colores_brillosos,  
    wedgeprops={'edgecolor': 'black'})
```

```

)

plt.title('Ventas por Categoría - Tienda 2', fontsize=14, weight='bold')
plt.ylabel('')
plt.tight_layout()
plt.savefig("figures/ventas_categoria2.png", dpi=300, bbox_inches='tight') #_
    ↪PNG de alta calidad
plt.show()

```



```

[203]: # Grafico de ventas mensuales - Tienda 3
colores_brillosos = ['#FF5733', '#33FF57', '#3357FF', '#FF33A1', '#FF8F33',
    ↪'#33FFFF', '#A633FF']
ventas_por_categoria = tienda3.groupby('Categoría del Producto')['Cantidad de_
    ↪cuotas'].sum()
ventas_por_categoria.plot.pie(

```

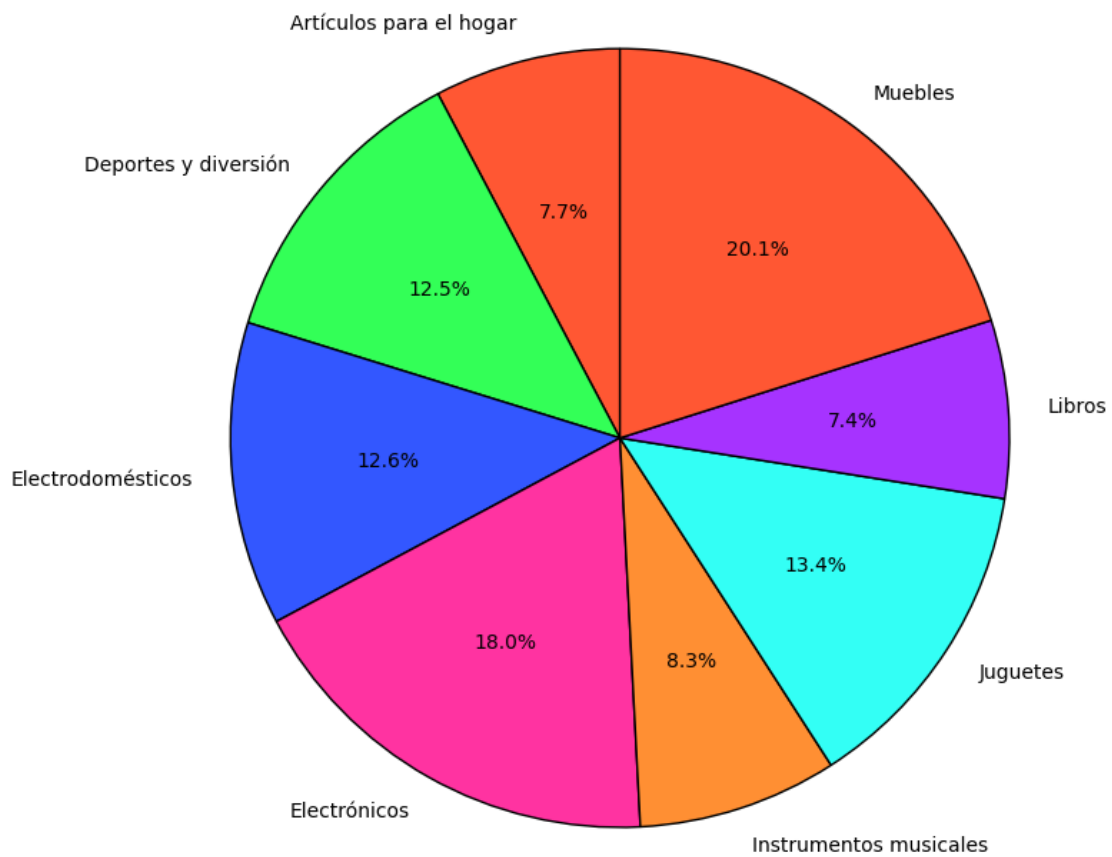
```

    autopct='%1.1f%%',
    startangle=90,
    figsize=(8,8),
    colors=colores_brillosos,
    wedgeprops={'edgecolor': 'black'}
)

plt.title('Ventas por Categoría - Tienda 3', fontsize=14, weight='bold')
plt.ylabel('')
plt.tight_layout()
plt.savefig("figures/ventas_categoria3.png", dpi=300, bbox_inches='tight') #_
    ↪PNG de alta calidad
plt.show()

```

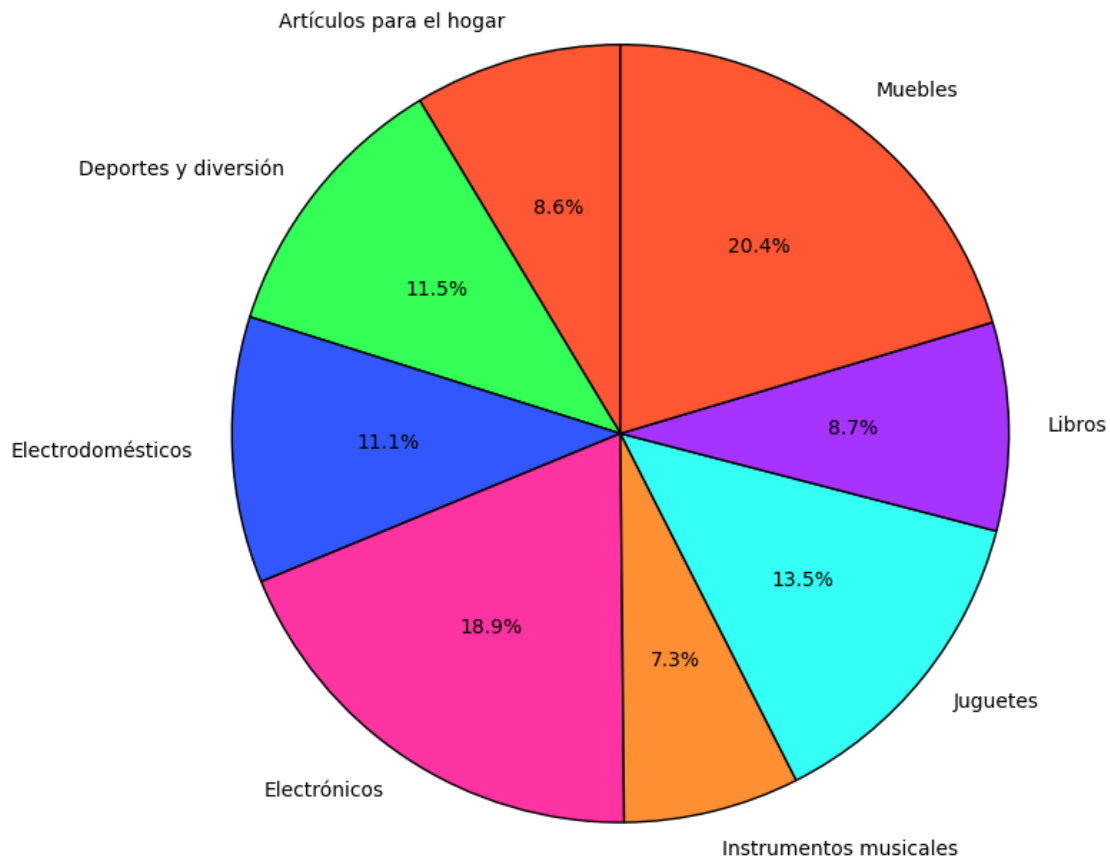
Ventas por Categoría - Tienda 3



```
[204]: # Grafico de ventas mensuales - Tienda 4
colores_brillosos = ['#FF5733', '#33FF57', '#3357FF', '#FF33A1', '#FF8F33',
↳ '#33FFF5', '#A633FF']
ventas_por_categoria = tienda4.groupby('Categoría del Producto')['Cantidad de_
↳ cuotas'].sum()
ventas_por_categoria.plot.pie(
    autopct='%1.1f%%',
    startangle=90,
    figsize=(8,8),
    colors=colores_brillosos,
    wedgeprops={'edgecolor': 'black'}
)

plt.title('Ventas por Categoría - Tienda 4', fontsize=14, weight='bold')
plt.ylabel('')
plt.tight_layout()
plt.savefig("figures/ventas_categoria4.png", dpi=300, bbox_inches='tight') #_
↳ PNG de alta calidad
plt.show()
```

Ventas por Categoría - Tienda 4



9 ¡Extra! Análisis del desempeño geográfico

```
[205]: # Centro del mapa: promedio de todas las coordenadas
lat_centro = tienda['lat'].mean()
lon_centro = tienda['lon'].mean()

m = fl.Map(location=[lat_centro, lon_centro], zoom_start=12)

# Agregar marcadores
for i, row in tienda.iterrows():
    fl.Marker(
        location=[row['lat'], row['lon']],
        popup=f"{row['Producto']} ({row['Categoría del Producto']})",
        tooltip=row['Producto']
```

```
).add_to(m)
```

```
[206]: m # El mapa se mostrará en el notebook interactivo (Colab).
```

```
[206]: <folium.folium.Map at 0x21c3be8c560>
```

10 Informe Final — Recomendación de Tienda para el Sr. Juan

Preparado para: Sr. Juan

10.1 Resumen ejecutivo

Este informe sintetiza el análisis de ventas, productos, costos y calificaciones de las cuatro tiendas de AluraStore Latam con el objetivo de recomendar en cuál debe concentrar el Sr. Juan sus esfuerzos comerciales. Tras evaluar ingresos totales, demanda por categorías, calificaciones de clientes, comportamiento de productos y costos de envío, recomendamos priorizar **Tienda 1** por su mayor potencial de rentabilidad y experiencia de cliente.

10.2 1. Alcance y metodología

- Datos analizados: ventas, cantidad por producto, categoría, precio, coste de envío y calificaciones de clientes.
- Fuentes: reportes internos de ventas y visualizaciones generadas en Jupyter/Colab.
- Técnicas: agregaciones por tienda y categoría, análisis de productos top/bottom, correlación simple entre precio y coste de envío, y comparación de calificaciones promedio.
- Limitaciones: no se incluyeron datos de estacionalidad ni costes operativos detallados (personal, alquileres). Recomendamos integrar esas variables en análisis futuros.

10.3 2. Hallazgos clave

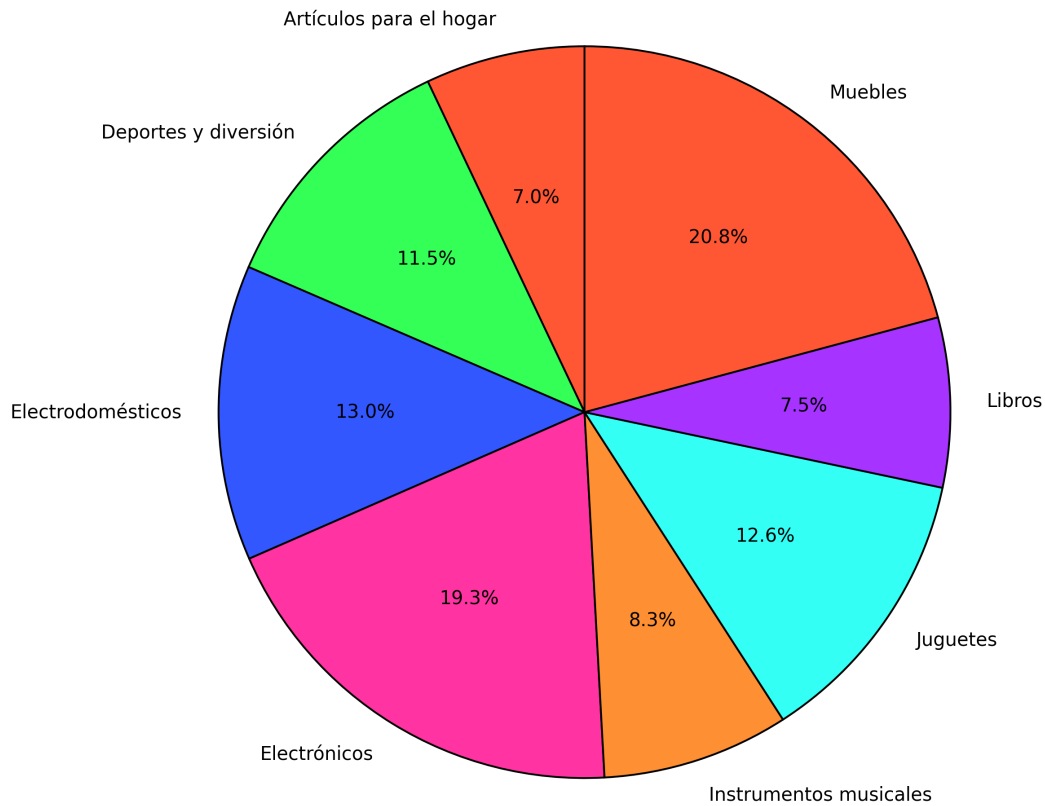
10.3.1 2.1 Ingresos totales por tienda

- Tienda 1 presenta los ingresos totales más altos, lo que sugiere mayor mercado y capacidad para generar margen.
- Otras tiendas exhiben ingresos inferiores y mayor variabilidad mensual.

10.3.2 2.2 Categorías de producto

- Tienda 1 muestra diversidad de categorías con alta demanda en las principales.
- Identificamos categorías consistentemente bajas en todas las tiendas que podrían considerarse para promociones o discontinuación. (*Anexo: Gráfico de ventas por categoría*)

Ventas por Categoría - Tienda 1

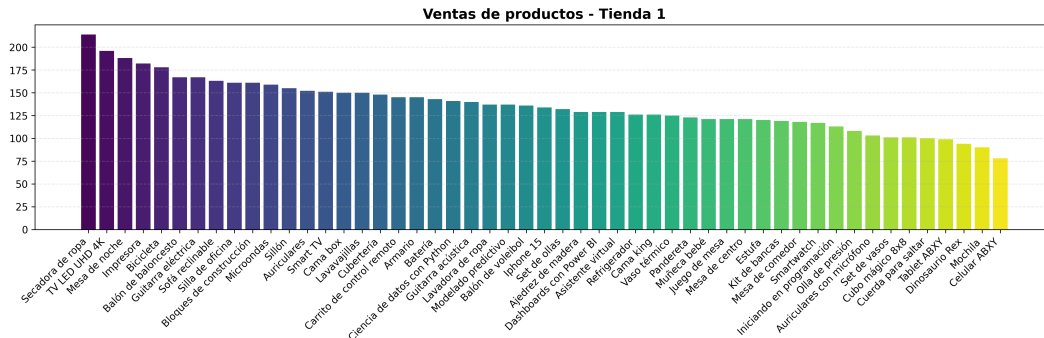


10.3.3 2.3 Calificaciones promedio

- Tienda 3 mantiene calificaciones promedio superiores, lo que favorece retención y conversión.
- Algunas tiendas presentan calificaciones bajas en subcategorías específicas — requiere atención a calidad/soporte.

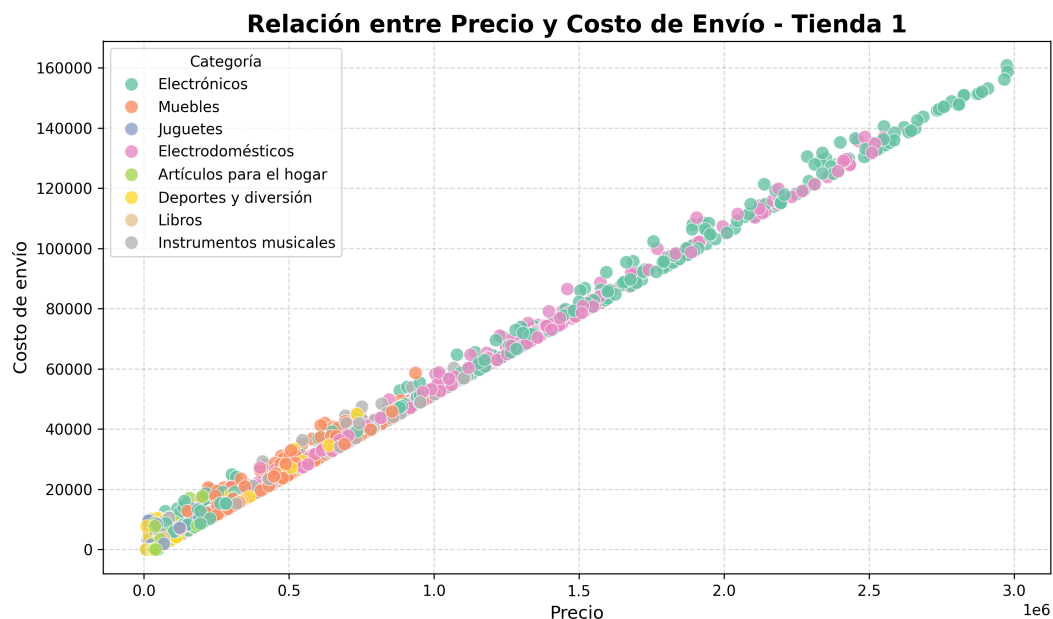
10.3.4 2.4 Productos más y menos vendidos

- Productos top de Tienda 1 son consistentes y generan flujo constante de ventas.
- Existencia de productos con rotación muy baja en otras tiendas; recomendable revisar políticas de inventario y promociones. (*Anexo: Gráfico de ventas por producto — Tienda 1*)



10.3.5 2.5 Coste de envío y relación con precio

- El coste de envío promedio es competitivo en Tienda 1.
- Aunque hay que tener en cuenta que el promedio de envío en Tienda 4 es el más bajo, sus ingresos y calificaciones son inferiores.
- Se observó una correlación positiva entre precio del producto y coste de envío (posible efecto por tamaño/peso/valor declarado), lo que puede afectar la decisión de compra. (*Anexo: Correlación precio vs. coste de envío*)



10.4 3. Conclusión y recomendación

Recomendamos que el Sr. Juan priorice sus ventas en **Tienda 1**, por las siguientes razones: - Mayores ingresos totales y menor volatilidad de ventas. - Catálogo con alta demanda y diversidad de categorías. - Mejores calificaciones promedio, que apoyan la fidelización. - Productos líderes con rotación constante. - Costes de envío competitivos que mejoran la propuesta de valor al cliente.

Las demás tiendas presentan brechas (ventas, calificaciones o costes) que las hacen menos atractivas como prioridad inmediata.

10.5 4. Acciones sugeridas (priorizadas)

1. Priorizar inventario y marketing en Tienda 1 (campañas de conversión sobre productos top, mejorar atención a cliente).
2. Ajustar precios y revisar la logística para minimizar el impacto del coste de envío en artículos de precio alto.
3. Ejecutar promociones para productos de baja rotación o evaluar su discontinuación.
4. Profundizar análisis con variables operativas (costes fijos, estacionalidad, disponibilidad de stock).
5. Monitorizar calificaciones por subcategoría y diseñar plan de mejora de experiencia (soporte y calidad).

10.6 5. Limitaciones y siguientes pasos

- Limitación: falta de análisis de costes operativos y estacionalidad.
- Sigüientes pasos: integrar datos operativos, ejecutar pruebas A/B de precios y envíos, y repetir el análisis tras 1–3 meses de acciones para validar impacto.

10.7 Anexos

- figures/ventas_categoria.png — Ventas por categoría
- figures/venta_productos_t1.png — Ventas por producto (Tienda 1)
- figures/corr_precio_costo.png — Correlación precio vs. coste de envío

Este informe resume las conclusiones derivadas de los análisis y visualizaciones en Jupyter/Colab y proporciona recomendaciones accionables para orientar la estrategia comercial del Sr. Juan.